# Image Retrieval System Based on EMD Similarity Measure and S-Tree

**Thanh Manh Le and Thanh The Van**

**Abstract** This chapter approaches the binary signature for each image on the base of the percentage of the pixels in each color image and builds a similar measure between the images based on EMD (earth mover's distance). Next, it aims to create S-tree in a similar measure EMD to store the image's binary signatures to quickly query image signature data. Then, from a similar measure EMD and S-tree, it provides an image retrieval algorithm and CBIR (content-based image retrieval). Last but not least, based on this theory, it also presents an application and experimental assessment of the process of querying image on the database system over 10,000 images.

**Keywords** CBIR • Image retrieval • EMD • S-tree • Signature • Signature tree

## 1 Introduction

It is difficult to find images in a large database of digital images. There are two main approaches for querying the images: querying the images based on the keyword TBIR (text-based image retrieval) [1] and those based on the content CBIR (content-based image retrieval) [1, 2].

In recent years, there have been considerable researches regarding CBIR, such as the image retrieval system based on color histogram [1–4], the similarity of the

T.M. Le
Hue University, Hue, Vietnam

T.T. Van (✉)
Center for Information Technology, HoChiMinh City University of Food Industry, HoChiMinh City, Vietnam
e-mail: thanhvt@cntp.edu.vn

J. Juang and Y.-C. Huang (eds.), *Intelligent Technologies and Engineering Systems*, Lecture Notes in Electrical Engineering 234, DOI 10.1007/978-1-4614-6747-2_17,

images based on histogram and the texture [5], and using the EMD distance in image retrieval [6–8].

This chapter aims to create the binary signature of an image and describe the distribution of image's colors by a bitstring with a given size. It also aims to query "similar images" in a large image database system efficiently. Additionally, two major targets are used to reduce the amount of storage space and speed up the query image on large database systems.

## 2 The Related Theory

### 2.1 S-Tree

S-tree [2, 9] is a tree with many branches that are balanced; each node of the S-tree contains a number of pairs $\langle \text{sig}, \text{next} \rangle$, where sig is a binary signature and next is a pointer to a child node. Each node root of the S-tree contains at least two pairs and at most $M$ pairs $\langle \text{sig}, \text{next} \rangle$, all internal nodes in the S-tree at least $m$ and at most $M$ pairs $\langle \text{sig}, \text{next} \rangle$, $1 \leq m \leq M/2$; the leaves of the S-tree contain the image's binary signatures sig, along with a unique identifier oid for those images. The S-tree height for $n$ signatures is at most $h = \lceil \log_m n - 1 \rceil$. The S-tree was built on the basis of *inserting* and *splitting*. When the node $v$ is full, it will be split into two.

### 2.2 EMD Distance

Setting $I$ as a set of suppliers, $J$ as a set of consumers, and $c_{ij}$ as the transportation cost from the supplier $i \in I$ to the consumer $j \in J$, we need to find out flows $f_{ij}$ to minimize the total cost $\sum_{i \in I} \sum_{j \in J} c_{ij} f_{ij}$ with the constraints [10] $f_{ij} \geq 0, \sum_{i \in I} f_{ij} \leq y_j$, $\sum_{j \in J} f_{ij} \leq x_i, i \in I, j \in J$. With $x_i$ as the provider's general ability $i \in I$, $y_j$ is the total need of the consumer $j \in J$. The feasible condition is $\sum_{j \in J} y_j \leq \sum_{i \in I} x_i$. The EMD distance [6, 7] is as follows: $\text{EMD}(x, y) = \left( \sum_{i \in I} \sum_{j \in J} c_{ij} f_{ij} \right) \Big/ \left( \sum_{i \in I} \sum_{j \in J} f_{ij} \right) = \left( \sum_{i \in I} \sum_{j \in J} c_{ij} f_{ij} \right) \Big/ \left( \sum_{j \in J} y_j \right)$

## 3 Building Data Structures and Image Retrieval Algorithms

### 3.1 Creating a Binary Signature of the Image Based on the Color Histogram

*Step 1*. Choose a standard color set $C = \{c_1, c_2, \ldots, c_n\}$ to calculate the color histogram of the images. To quantify the image $I$ in order to retain only the dominant colors $C_I = \left\{c_1^I, c_2^I, \ldots, c_{n_I}^I\right\}$, the color histogram vector of image $I$ is $H_I = \left\{h_1^I, h_2^I, \ldots, h_{n_I}^I\right\}$.

*Step 2*. Calculate the color histogram vector standardizes $H = \{h_1, h_2, \ldots, h_n\}$, where $h_i = h_j^I \Big/ \sum_j h_j^I$ if $c_i \in C \cap C_I$, otherwise $h_i = 0$.

*Step 3*. Each color $c_j^I$ will be described into a bitstring $b_1^j b_2^j, \ldots, b_m^j$. The binary signature of the image $I$ will be sig $= \mathrm{B}^1\mathrm{B}^2 \ldots \mathrm{B}^n$, $B^j = b_1^j b_2^j \ldots b_m^j$, in which $b_i^j = 1$ if $i = \lceil h_i \times m \rceil$, otherwise $b_i^j = 0$.

### 3.2 Measuring Similar Image Based on EMD Distance

The weight of the component $B_I^j = b_1^j b_2^j \ldots b_m^j$ is $w_I^j = w\left(B_I^j\right) = \sum_{i=1}^{m}\left(b_i^j \times (i/m) \times 100\right)$; the weight vector of the image $I$ will be $W_I = \left\{w_I^1, w_I^2, \ldots, w_I^n\right\}$. $J$ is the image that we need to calculate the similarity with the image $I$, so we need to minimize the cost $\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij} f_{ij}$, and $F = \left(f_{ij}\right)$ is the matrix of color distribution flows from $c_I^i$ to $c_J^j$ and $D = \left(d_{ij}\right)$ is the Euclidean distance matrix in the RGB color space from $c_I^i$ to $c_J^j$. The similarity between two images $I$ and $J$ based on the EMD distance will minimize the value $\mathrm{EMD}(\mathrm{I}, \mathrm{J}) = \min_{\mathrm{F}=\left(\mathrm{f_{ij}}\right)} \left(\sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{n}} \mathrm{d_{ij}f_{ij}}\right) \Big/ \sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{n}} \mathrm{f_{ij}}$, with $\sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{n} w_I^i, \sum_{j=1}^{n} w_J^j\right)$

### 3.3 Creating S-Tree Based on EMD Distance

***Algorithm1.*** `Gen-Stree(S, Root)`
*Step 1*. `v = Root;`
`If S = ∅ then STOP;`

```
        Else Choosing <sig,oid> ∈ S and S = S \ <sig,oid>;
        To go Step 2;
Step 2. If v is leaf then
        begin
             v = v ⊕ <sig,oid>; UnionSig(v);
             If v.count > M then SplitNode(v);
             To go back Step 1;
        end
        Else
        begin
             EMD(SIG0→sig,sig)=min{EMD(SIGi→sig,sig) |
             SIGi∈ v};
             v = SIG0→next; To go back Step 2;
        End
```

Splitting the node $v$ based on $\alpha - seed$ and $\beta - seed$ in [2, 9] is done as follows:

```
Algorithm2. SplitNode(v)
Create the nodes vα and vβ contains α − seed and β − seed;
For (SIGi ∈ v) do
Begin
    If (EMD(SIGi→sig, α − seed) < EMD(SIGi→sig, β − seed)) then
       vα = vα ⊕ SIGi;
    Else vβ = vβ ⊕ SIGi;
    sα = ⋃ sig_i^α, with sig_i^α ∈ vα;  sβ = ⋃ sig_i^β, with sig_i^β ∈ vβ;
    v_parent = v_parent ⊕ sα; v_parent = v_parent ⊕ sβ;
    If (v_parent.count > M) then SplitNode(v_parent);
End.
Procedure UnionSig(v)
Begin
s = ⋃ sig_i, with sig_i ∈ v;
If (v_parent != null) then
begin
   SIG_v = {SIG_i | SIG_i → next = v, SIG_i ∈ v_parent}; v_parent → (SIG_v → sig) = s;
   UnionSig(v_parent);
end
End.
```

## 3.4 The Image Retrieval Algorithm Based on S-Tree and EMD Distance

```
Algorithm3. Search-Image-Sig(sig, S-tree)
v = root; SIGOUT = ∅ Stack = ∅ Push(Stack, v);
```

```
while(not Empty(Stack)) do
begin
  v = Pop(Stack);
  If(v is not Leaf) then
  begin
        For(SIG_i ∈ v and SIG_i→sig ∧ sig = sig) do
              EMD(SIG_0→sig,sig)=min{EMD(SIG_i→sig,sig)|
              SIG_i∈v};
        Push(Stack, SIG_0→next);
      end
      Else SIGOUT = SIGOUT ∪ {<SIG_i→sig,oid_i>|SIG_i∈v};
end
return SIGOUT;
```

# 4 Experiments

## *4.1 Model Application*

***Phase 1:*** *Perform Preprocessing* (Fig. 1)

*Step 1*. Quantize images in the database and convert to a color histogram.
*Step 2*. Convert the color histogram of the image in the form of binary signatures.
*Step 3*. Respectively calculate the similarity measure EMD distance of the image signatures and insert into the S-tree.

***Phase 2:*** *Implementation Query*

*Step 1*. For each query image, calculate the color histogram and convert into binary signatures.
*Step 2*. Perform binary signature query on the S-tree consisting of the image signature, it is possible to find similar images at the leaves of the S-tree through the EMD measure.
*Step 3*. After finding similar images, conduct arrangement of similar levels from high to low and make the title match with the images arranged on the basis of similarity EMD distance.

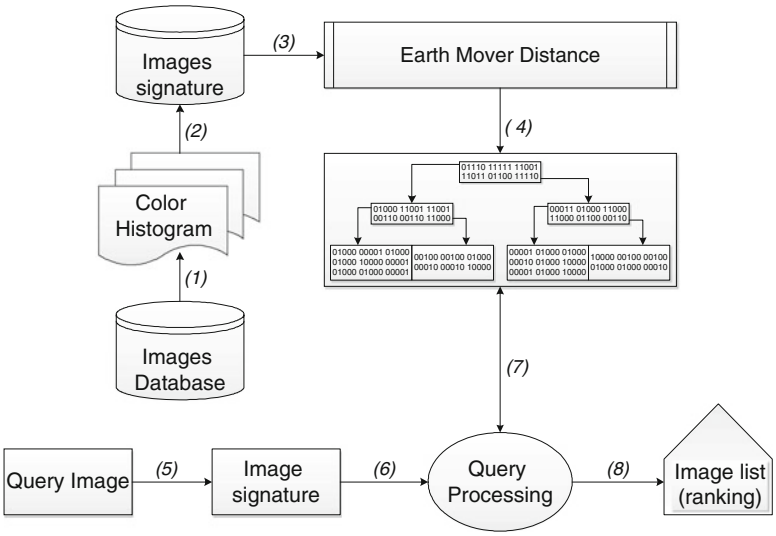


**Fig. 1** Model image retrieval system

**Fig. 2** A sample result of the process query image in image database over 10,000 images

## *4.2 The Experimental Results*

Each image will calculate the color histogram based on 16 colors: black, silver, white, gray, red, orange, yellow, lime, green, turquoise, cyan, ocean, blue, violet, magenta, and raspberry (Figs. 2 and 3).

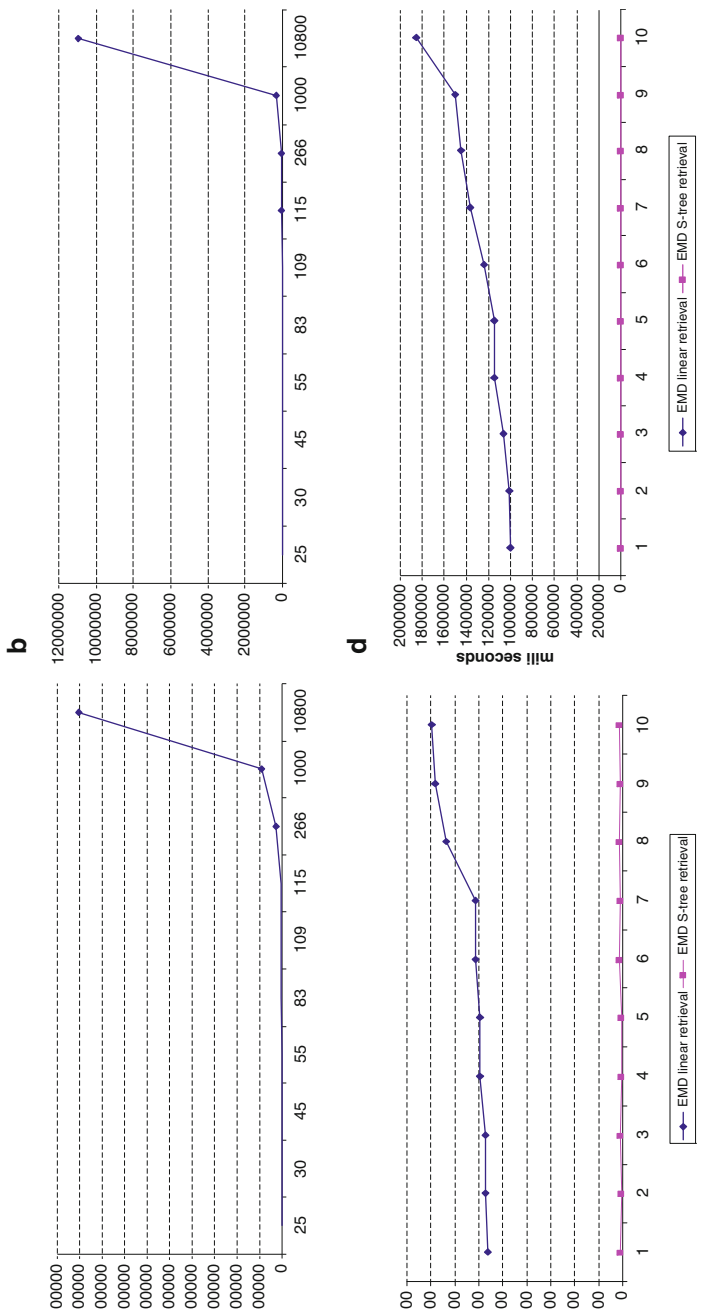


**Fig. 3** (**a**) Number of comparisons to create S-tree. (**b**) The time to create S-tree. (**c**) Number of comparisons to query image in database over 10,000 images. (**d**) The time to query image in database over 10,000 images

## 5 Conclusion

This chapter creates algorithms in order to speed up the retrieval of similar images based on the image's binary signatures and then designs and implements the image retrieval model CBIR. As can be seen from the experiment, it takes a long time to create the S-tree from the image's binary signature, but the retrieval of the image that relies on the S-tree will be a lot faster than a linear search method based on EMD. However, using EMD to calculate the distribution of the image's colors will result in inaccuracy in the case of the images with the same percentage of color pixels, but the color distribution location does not correspond to each other. The next development will assess the similarity of the image through EMD distance with location distribution of the percentage of color pixels and compare the objects in the contents of the image to increase accuracy when querying the similar images.

## References

1. Neetu Sharma S, Paresh Rawat S, Jaikaran Singh S (2011) Efficient CBIR using color histogram processing. Signal Image Process Int J 2(1):94–112
2. Nascimento MA, Tousidou E, Chitkara V, Manolopoulos Y (2002) Image indexing and retrieval using signature trees. Data Knowl Eng 43(1):57–77
3. Abuhaiba ISI, Salamah RAA (2012) Efficient global and region content based image retrieval. Int J ImageGraphics Signal Process 4(5):38–46
4. Yu J, Amores J, Sebe N, Radeva P, Tian Q (2008) Distance learning for similarity estimation. IEEE Trans Pattern Anal Mach Intell 30(3):451–462
5. Kavitha C, Babu Rao M, Prabhakara Rao B, Govardhan A (2011) Image retrieval based on local histogram and texture features. Int J Comput Sci Inf Technol 2(2):741–746
6. Rubner Y, Tomasi C, Guibas LJ (1998) A metric for distributions with applications to image databases. In: Proceedings of the IEEE international conference on computer vision, Bombay, India, 4–7 January 1998, pp 59–66
7. Abdelkhalak B, Zouaki H (2011) EMD similarity measure and metric access method using EMD lower bound. Int J Comput Sci Emerg Technol 2(6):323–332
8. Hurtut T, Gousseau Y (2008) Francis Schmitt: adaptive image retrieval based on the spatial organization of colors. Comput Vis Image Underst 112(2):101–113
9. Chen Y, Chen Y (2006) On the signature tree construction and analysis. IEEE Trans Knowl Data Eng 18:1207–1224
10. Konstantinidis K, Gasteratos A, Andreadis I (2005) Image retrieval based on fuzzy color histogram processing. Sci Direct Optics Commun 248(4–6):375–386